TMS-120

U S B S e r i a l E E P R O M P r o g r a m m e r

User's Manual

Revision A – DRAFT 1

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

Table of Contents

1 INTRODUCTION 8

1.1 WHAT IS THE TMS-120 8
1.2 SPECIFICATIONS 8
1.3 APPLICATIONS FOR THE TMS-120 9
1.4 HOW THE TMS-120 WORKS 9

2 USB INTERFACE 10

2.1 CONNECT THE TMS-120 TO THE USB PORT 10
2.2 INSTALLING THE USB PORT DRIVER SOFTWARE 10
2.3 COM PORT IDENTIFICATION 13
2.4 CHANGING THE COM PORT SELECTION 14

3 "EEPROM TOOL" WINDOWS® SOFTWARE 16

3.1 INSTALLATION 16
3.1.1 From the CD-ROM 16
3.1.2 From the Internet 16
3.2 CONNECTING THE TMS-120 TO YOUR EEPROM 16
3.2.1 Interface Module Installation 16
3.2.2 EEPROM Interface Options 18
3.2.3 I2C (24C' Series) Interface Cable 19
3.2.4 SPI (25C' Series) / Microwire (93C' Series) Interface Cable 20
3.2.5 SOIC Test Clip (Optional) 21
3.2.6 Interface Cable Data 22
3.2.7 Custom Interface Solution 22
3.3 SOFTWARE CONFIGURATION 24
3.3.1 EEPROM Tool Software Configuration 24
3.3.2 Communications Port 24
3.3.3 EEPROM Configuration 24
3.3.4 Device type 24
3.3.5 Write Buffer Size 25
3.3.6 Address Pins A0, A1, and A2 (24C' Series Only) 25
3.3.7 Organization (93C' Series Only) 25
3.3.8 Data File Folder Selection 26
3.3.9 Exiting the Configuration Dialog 26

4 USING THE TMS-120 27

4.1 ATTACHING THE TMS-120 TO A TARGET EEPROM 27
4.1.1 Typical I2C (24C' Series) Connection 28
4.1.2 Typical SPI (25C' Series) Connection 29
4.1.3 Typical Microwire (93C' Series) Connection 31

5 USING THE EEPROM TOOL 33

5.1 READING DATA FROM THE EEPROM 35
5.2 WRITING DATA TO A SINGLE EEPROM CELL 35

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.3 Verifying EEPROM Contents 36
5.4 Erasing an EEPROM 36
5.5 Exporting Data to Disk File 37
5.6 Coping EEPROMs 37
5.7 Clearing the Data Window and Initializing the Program 37
5.8 Importing Data from Disk File 37
5.9 Writing an Entire EEPROM at Once 38
5.10 Printing Data 39
5.11 The SPI EEPROM Status Register 40

6 TROUBLESHOOTING 42

6.1 Verify Interface (Data Communications Test) 42
6.2 The Status Bar 42
6.2.1 General I/O Errors 42
6.2.2 Communications Errors 43
6.2.3 Operational Errors 44

7 COMMUNICATION PROTOCOL 47

7.1 Protocol Basics 47
7.2 General Command Protocol 47
7.3 Status Codes 49
7.4 Command 0x10 - Read Serial EEPROM Data 51
7.5 Command 0x11 - Write Serial EEPROM Data 53
7.6 Command 0x12 - Read Status Register of SPI EEPROM 53
7.7 Command 0x13 - Write Status Register of SPI EEPROM 55

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

TMS-120 User's Manual

End User License Agreement (EULA): The use of this device (TMS-120) and the accompanying PC application (EEPROM Tool) constitutes acceptance of this agreement on the part of the consumer. The term "software" in this EULA jointly represents both the internal TMS-120 firmware and the PC application, EEPROM Tool, and associated computer files. The software included with this device is protected by copyright law and international treaties. Unauthorized reproduction or distribution of this software, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law. By installing, copying, or otherwise using this software, the user agrees to be bound by the terms and conditions of this license agreement. This software is the property of Triangle Micro Solutions, Inc. The user may not disassemble, de compile, or modify any part of the application or its associated files. The user may not sell, rent, lend or otherwise distribute this software on a stand alone basis or as part of a collection, or as a part of any product or service for resale unless you have previously formed a license agreement with Triangle Micro Solutions, Inc. This software is supplied with **NO WARRANTIES**. Triangle Micro Solutions, Inc. expressly disclaims any warranty for the software. This software and related documents are provided "AS IS" without warranty of any kind, expressed or implied. This includes implied warranties of fitness for a particular purpose. All risk arising out of use of this software remains with the user. By using this software, this user agrees that in no event, Triangle Micro Solutions, Inc. can be held responsible or liable for damages whatsoever. This includes, without limitation, damages for loss of business profits, loss due to business interruption, loss of business information, or any other pecuniary loss, arising out of the use of or the inability to use this software.

Warranty: This product is covered for one year from the date of sale for manufacturing defects in the TMS-120 module only. The interface cables and the software CD-ROM are not covered. If the TMS-120 module fails to operate within this one year period, it will be repaired or replaced at the discretion of Triangle Micro Solutions, Inc. The consumer is responsible for shipping charges. The consumer must contact Triangle Micro Solutions, Inc. and inform us prior to returning any merchandise.

Statement of Non Compliance: This device generates and uses radio frequency (RF) energy and could emit sufficient levels of RF energy to interfere with other electronic devices. An increase in emissions is possible when this device is connected to external equipment. This device is intended for use in a laboratory environment only. It has not been tested for compliance with the limits for a class A computing device pursuant to subpart J of part 15 of the FCC regulations. It is the user's responsibility to take corrective action if interference occurs.

USE THIS DEVICE AT YOUR OWN RISK!

Please read this documentation before using the TMS-120. While the TMS-120 can be a useful tool, it must also be understood that the TMS-120 could render target devices non-functional. If you are using the TMS-120 to write data to EEPROM memory devices, be sure to make a back up copy of any data that is critical for the operation of the device. It is further recommended that redundant back up copies of critical data be maintained on a separate storage system (i.e.: floppy disk or network drive). Triangle Micro Solutions, Inc. is not responsible for any malfunction or damages caused by the TMS-120 regardless of how the damage occurs. This device was not designed for use in life support appliances, devices, systems or in any application where a malfunction may reasonably be expected to cause personal injury or loss of life.

Use the TMS-120 at your own risk!

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

For technical support and sales information, contact Triangle Micro Solutions, Inc. at:

Triangle Micro Solutions, Inc.

25 Oak Hollow Court
Louisburg, North Carolina 27549
USA

Contact by email: sales@trianglemicro.com
support@trianglemicro.com

On-line visit: www.TriangleMicro.com

By Phone: (919) 495-0622

User Manual Revision: A (March 2007)

Supports: TMS-120 Interface Module Firmware Rev. 1.0.0 (or greater)
EEPROM Tool Software Version 3.4.0 (or greater)

Windows™ is a trademark of Microsoft Corporation
I^2C™ is a trademark of Philips Corporation
SPI™ is a trademark of Motorola
Microwire™ is a trademark of National Semiconductor Corp.

Please consult the README.TXT file on the accompanying CD-ROM for any changes or deviation from the information documented in this manual. We reserve the right to make changes as needed without notification.

PRELIMINARY

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

1 Introduction

1.1 What is the TMS-120

The TMS-120 is a *Serial EEPROM Programmer*. A Serial EEPROM (Electrically Erasable Programmable Read Only Memory) is a type of non-volatile memory used in many different types of digital electronic devices. The TMS-120 consists of an interface module, two interface cables, and the "EEPROM Tool" software for Windows® 98 (or greater). Together, these items give your computer the ability to program and read most serial EEPROM devices in use today. The TMS-120 is designed to program EEPROM devices in circuit without having to remove them. Under most circumstances, this eliminates the need to de-solder an EEPROM from a printed circuit board.

1.2 Specifications

The following specifications apply to the TMS-120 and the EEPROM Tool software:

PRELIMINARY

Interface Module Power:	Power is supplied by the USB port.
EEPROM Devices Supported:	***I^2C EEPROMs Supported*** 24C00, 24C01, 24C02, 24C04, 24C08, 24C16, 24C32, 24C64, 24C65, 24C128, 24C256 and 24C512. ***SPI EEPROMs Supported*** 25C040, 25C080, 25C160, 25C320, 25C640 ***Microwire EEPROMs Supported*** 93C06, 93C46, 93C56, 93C66, 93C76, 93C86 (8 or 16 bit organization) Where 'C' could be 'AA', 'C', or 'LC'.
Software:	Requires Windows® 95, 98, 98SE, 2K, ME, or XP. ***NOTE**: EEPROM Tool has not been tested with the Vista version of Windows.*
Disk Space:	8 MB
Processor:	Pentium III or better recommended
USB Port:	Driver Supplied on CD-ROM
Mechanical Dimensions:	3" x 1 5/8" x 3/4"
Pointing Device:	A mouse is recommended for some software operations.

Table 1.1: Specifications

Copyright © 2007 - Triangle Micro Solutions, Inc.

1.3 Applications for the TMS-120

The TMS-120 has many applications, here are just a few:

- Product initialization and configuration (i.e. serial numbers, calibration data, or other manufacturing data can be loaded into your product on the factory floor).
- Hardware / Software Integration Testing.
- Embedded system product development.
- Modifying security codes for car radios.
- TV / Satellite receiver repair.
- Computer repair and maintenance (Desktops / Laptops).
- Computer peripheral repair and maintenance (LAN cards, routers, printers, monitors, etc.)

1.4 How the TMS-120 Works

The interface module contains a small microcontroller, which connects to your computer's USB port. The microcontroller also has an I^2C bus port and SPI/Microwire port accessed via an RJ45 connector, which is located on the opposite end of the interface module. The firmware inside the microcontroller receives commands from the EEPROM Tool software through the USB port. The commands are translated into serial data streams, and sent to the serial EEPROM. The commands contain read or write instructions for the EEPRPOM memory. The serial bus then relays data back to the microcontroller and the microcontroller in turn sends the data back to the EEPROM Tool software over the USB port.

PRELIMINARY

NOTE

The TMS-120 does ***NOT*** apply power to EEPROM devices. It is required that the EEPROM be in a working circuit with power applied before it can be programmed, read or erased with the TMS-120.

Copyright © 2007 - Triangle Micro Solutions, Inc.

2 USB Interface

Before the TMS-120 can be used, the USB driver must be installed. This driver is included on the CD-ROM that was supplied with the TMS-120 or can be downloaded from www.TriangleMicro.com/TMS-120.htm. The USB driver creates a virtual RS-232 (COM) port. The COM port number must be known when setting up the EEPROM Tool software.

2.1 Connect the TMS-120 to the USB Port

The TMS-120 is supplied with a USB cable, which is terminated with a USB-A male connector on one end and a USB-B male connector on the other end. The USB cable provides both power and communications to the TMS-120. Insert the USB-B Male end into the receptacle on the TMS-120. Insert the USB-A Male end of the cable into a spare USB port on your computer. If this is the first time that the TMS-120 has been connected to this USB port, the PC will indicate that there is new hardware found and will try to install the USB driver. Follow the instructions in the next section for driver installation.

2.2 Installing the USB Port Driver Software

Insert the CD-ROM supplied with the TMS-120 into your CD-ROM drive, or if you have downloaded driver files from www.TriangleMicro.com, be prepared to point Found New Hardware Wizard to the location of the driver files. When the Found New Hardware Wizard window is displayed, select "No, not this time" and select "Next".

Found New Hardware Wizard
Welcome to the Found New Hardware Wizard
Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission).
Read our privacy policy
Can Windows connect to Windows Update to search for software?
Yes, this time only
Yes, now and every time I connect a device
No, not this time
Click Next to continue.
< Back
Next >
Cancel

Figure 2-1: Found New Hardware Wizard - Welcome

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

In the next dialog window, you will be asked how to install the driver software. Choose "Install the software automatically (recommended)" and click "Next" to continue.

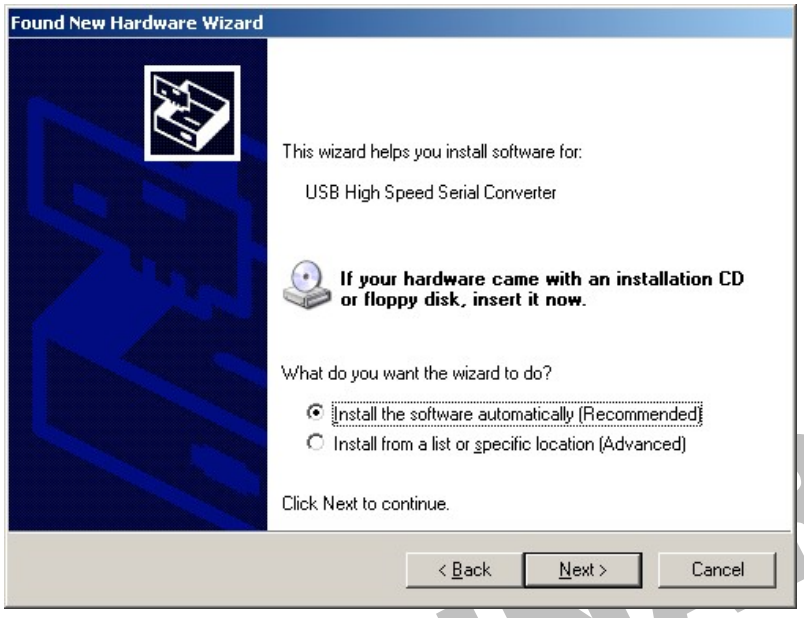

Found New Hardware Wizard
This wizard helps you install software for:
USB High Speed Serial Converter
If your hardware came with an installation CD or floppy disk, insert it now.
What do you want the wizard to do?
Install the software automatically (Recommended)
Install from a list or specific location (Advanced)
Click Next to continue.
< Back
Next >
Cancel

Figure 2-2: Found New Hardware Wizard - Location

The next dialog window will indicate that the "USB High Speed Serial Converter" has not passed Windows® Logo testing. Select "Continue Anyway".

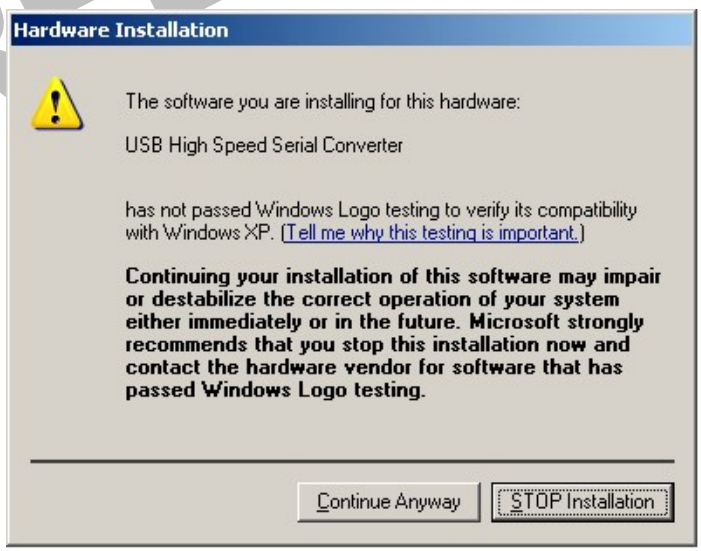

Hardware Installation
The software you are installing for this hardware:
USB High Speed Serial Converter
has not passed Windows Logo testing to verify its compatibility with Windows XP. (Tell me why this testing is important.)
Continuing your installation of this software may impair or destabilize the correct operation of your system either immediately or in the future. Microsoft strongly recommends that you stop this installation now and contact the hardware vendor for software that has passed Windows Logo testing.
Continue Anyway
STOP Installation

Figure 2-3: Hardware Installation

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

The wizard will then install the USB driver.

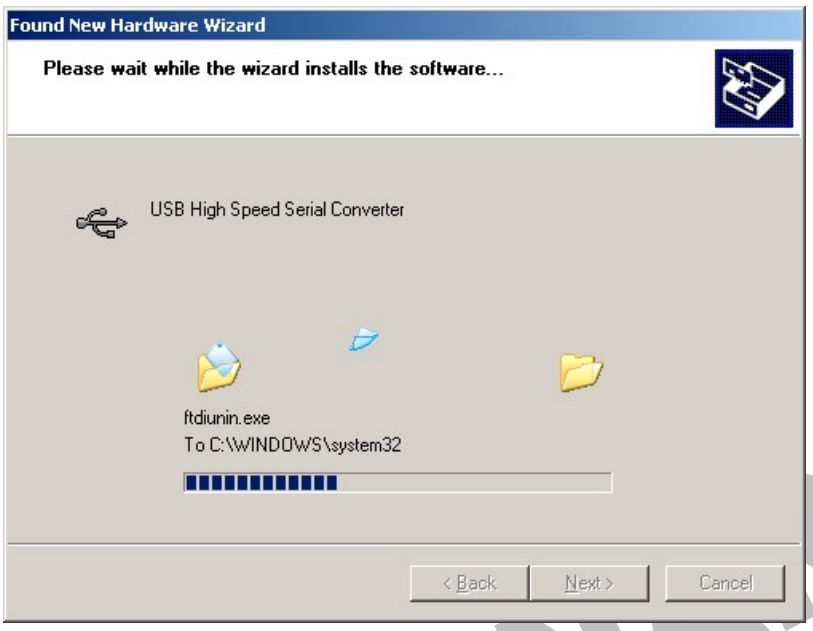


Figure 2-4: Found New Hardware Wizard – Installation

The wizard completes the installation of the driver.

Found New Hardware Wizard
Completing the Found New Hardware Wizard
The wizard has finished installing the software for:
USB High Speed Serial Converter
Click Finish to close the wizard.
< Back
Finish
Cancel

Figure 2-5: Found New Hardware Wizard – Completion

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

2.3 COM Port Identification

Open the *Control Panel* (In Windows XP, select *Start* → *Settings…* → *Control Panel*). Then from the *Control Panel* window, select *System* and select the *Hardware* tab. Finally, click on the Device Manager button and a window similar to the one shown in the figure below should appear. Expand *Ports* to see a list of COM and LPT ports. The TMS-120 port is identified in the figure below. The COM port is shown here as COM4. Your actual port may be different. Note the COM port selected in your installation because it will be needed to configure the EEPROM Tool application software.

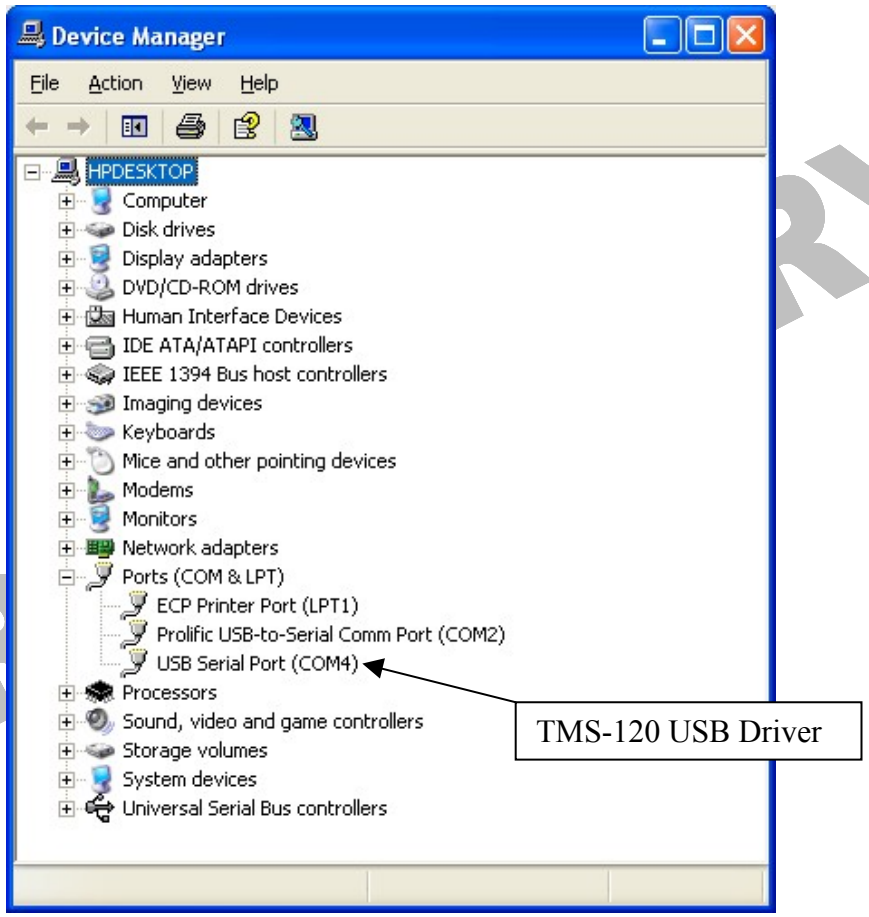


Figure 2-6: Device Manager Window

Copyright © 2007 - Triangle Micro Solutions, Inc.

2.4 Changing the COM Port Selection

The EEPROM Tool application software supports COM1 – COM7. If the installation selected a COM Port greater than 7, or if a different COM port is desired due to a conflict, etc. follow these instructions to change the COM port selection. Right-click on the driver in the Device Manager window and select Properties.

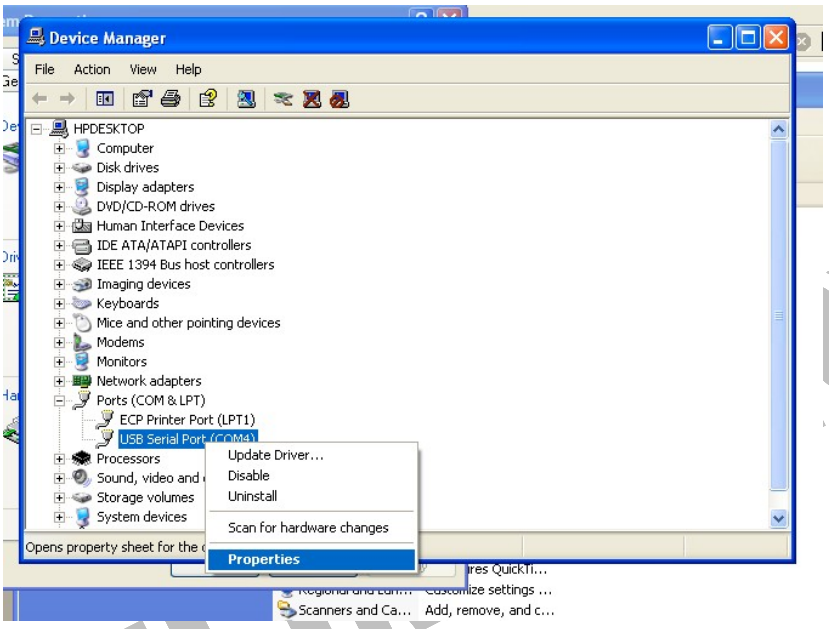


Figure 2-7: COM Port Properties

From the USB Serial Port Properties dialog box, select the *Advanced* button.

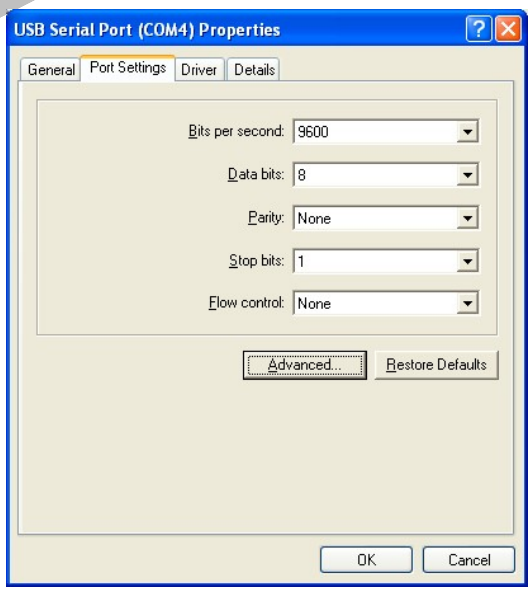


Figure 2-8: USB Serial Port Properties

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

In the Advanced Settings window, there is a selection for COM Port Number. Select the desired COM port and click the OK button to save the selection.

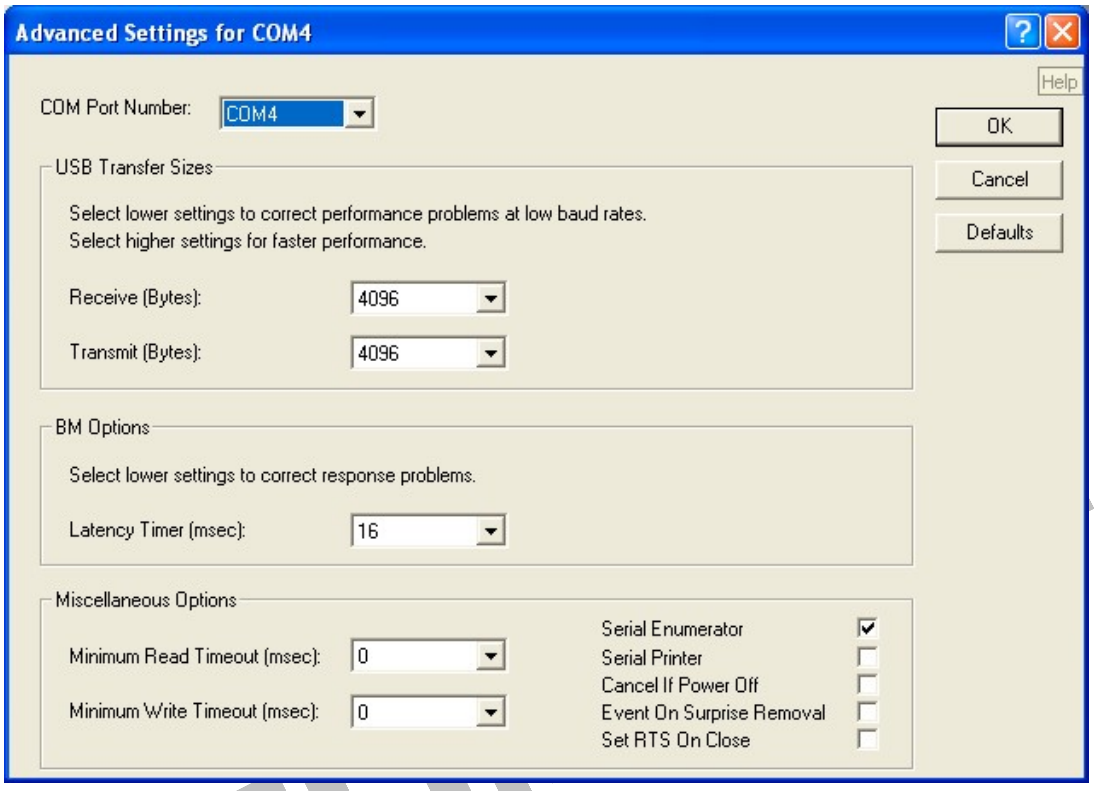


Figure 2-9: Advanced Settings Dialog Box

PRELI

Copyright © 2007 - Triangle Micro Solutions, Inc.

3 "EEPROM Tool" Windows® Software

The EEPROM Tool software must be installed on you computer before you can use the TMS-120. Please close all applications running on your computer before installing the EEPROM Tool software.

3.1 Installation

3.1.1 From the CD-ROM

To install the software from CD-ROM, insert the CD-ROM into the appropriate drive. Select ***Start*** | ***Run*** from the Windows task bar. Enter "X:SETUP.EXE" in the "open" field, then press the "ok" button. Be sure to replace the 'X' shown in the example with the actual drive letter for your CD-ROM. Follow the instructions provided by the install wizard. Allow the install wizard to reboot your computer.

3.1.2 From the Internet

The EEPROM Tool software may also be obtained or updated at anytime over the Internet. New EEPROM Tool updates may become available from time to time and are always free. Updates downloaded from the Internet will be in a zip file. You will need a utility such as **Winzip®** to decompress or "unzip" the files. Unzip the files into a temporary directory (i.e.: C:\Temp) and then install the software from that directory using the instructions in section 2.2 above. Use the path to the temporary directory instead of the CD-ROM. Be sure to view the readme.txt file first for the latest information regarding installing or using EEPROM Tool.

Use the URL below to get software updates.

http://www.trianglemicro.com/TMS-120.htm

Use the URL below to obtain the **Winzip®** utility.

http://www.winzip.com

3.2 Connecting the TMS-120 to your EEPROM

3.2.1 Interface Module Installation

The Interface Module is shown in figure 3.1. The module is housed in a molded plastic enclosure with a 25 pin D-sub connector (female) connector at one end and an 8-pin RJ-45 connector at the other end. Connect the 25 D-sub connector to your PC's serial port. Most modern PC's do not have a 25-pin serial (RS-232) port available. If your PC does not have a 25-pin serial port, a 9-to-25-pin converter will be needed. These are available from Triangle Micro Solutions (order P/N TMS-DB9).

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

FIGURE MISSING

Figure 3.1 - The TMS-120 Interface Module

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.2.2 EEPROM Interface Options

You have several options when it comes to connecting the TMS-120 to the target EEPROM. The TMS-120 is supplied with two interface cables. The *I^2C Interface Cable* is designed to connect the TMS-120 to I^2C (24C' Series) EEPROM devices. The *SPI/Microwire Interface Cable* is designed to connect the TMS-120 to SPI or Microwire EEPROM devices. Both cables are approximately 5 feet long and are terminated with a RJ-45 connector on one end and have loose wires terminated with pin receptacles on the other end. The RJ-45 end is inserted into the TMS-120 interface module. The wires terminated with pin receptacles can be inserted onto test hooks (supplied) or the SOIC test clip (optional). The test hooks are double hook micro grabbers suitable for grabbing the pins of DIP or SOIC packages. Each of the loose wires has a color-coded end that matches the color of the test hook that it connects to. When configuring the cables, be sure to match the wire end color with the test hook color.

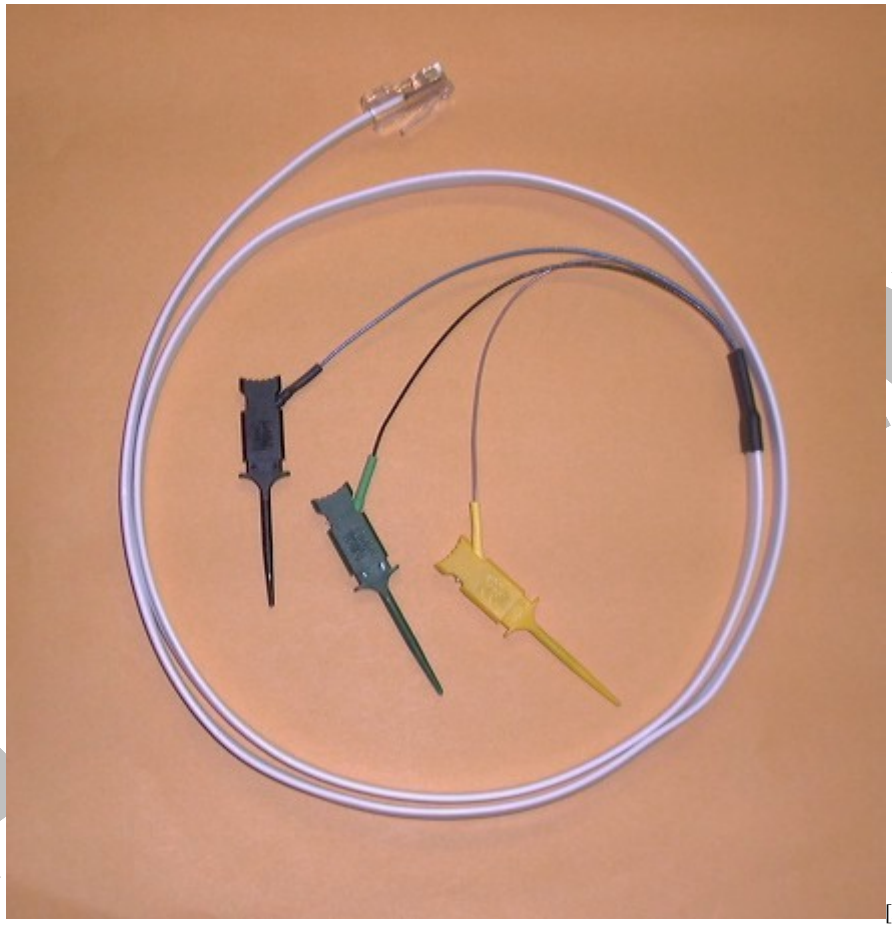
[APS1]

Figure 3.1-TMS-120 Interface Cables and Test Hooks

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.2.3 I2C (24C' Series) Interface Cable

Figure 3.3 below shows the I^2C Interface Cable with the green, yellow and black test hooks attached. In this configuration, the cable is ready to attach to an I^2C EEPROM device while in circuit. Please review section 5.1.1 of this document before attaching the cable to your target EEPROM.



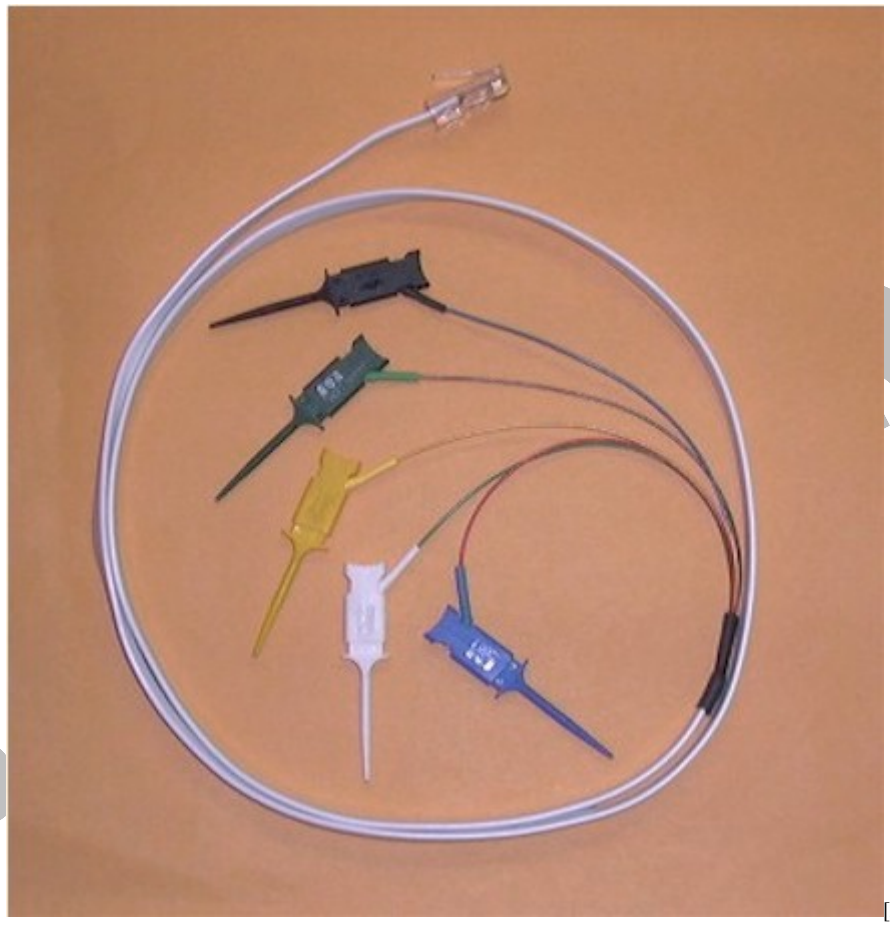
[APS2]

Figure 3.2 - I^2C Interface Cable with Test Hooks Attached

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.2.4 SPI (25C' Series) / Microwire (93C' Series) Interface Cable

Figure 3.3 below shows the SPI / Microwire interface cable with all of the micro-grabbers attached. In this configuration, the cable is ready to attach to an SPI or Microwire EEPROM device while in circuit. Please review section 5.1.2 or section 5.1.3 of this document before attaching the cable to your target EEPROM.



[APS3]

Figure 3.3 - SPI / Microwire Interface Cable with Test Hooks Attached.

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.2.5 SOIC Test Clip (Optional)

The SOIC Test Clip offers a more convent way to connect a TMS-120 to a SOIC 8-pin IC package. For high volume application, the test clip provides a faster means of connecting the TMS-120 to the EEPROM. Test clips can be purchased directly from Triangle Micro Solutions (order P/N TMS-SOIC8). To use the test clips, remove the Test Hooks from the cable that you wish to use and connect the pin receptacle on the appropriate test clip pin. Verify that you have done this correctly by checking the data sheet for the specific EEPROM being programmed.

[APS4]

Figure 3.4 - TMS-120 Cable with SOIC Test Clip Attached

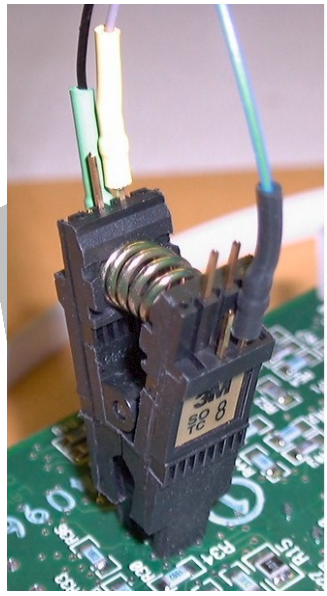


Figure 3.5 - SOIC Test Clip In Use

Figure 3.4 shows a TMS-120 I^2C Interface Cable with the Test Hooks removed and the optional SOIC Test Clip in place. In figure 3.5, the Test Clip is shown connected to an EEPORM in circuit. In this example, the EEPROM is a 24C series (I^2C) EEPROM, but the Test Clip may be used with SPI or Microwire devices as well. Consult the device data sheet for your specific device for pin configuration. Note that the SOIC Test Socket pins are on two different levels. This can make pin connections tricky, so check these thoroughly before use.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.2.6 Interface Cable Data

Table 3.3 below identifies the function, wire color, and test hook color for both of the interface cables supplied with the TMS-120. See section 5 for photos showing wire function and how they are typically connected to EEPROM devices in real applications.

RJ-45 Pin No.	Wire Color	Wire Function	
		I^2C Cable	**SPI / Microwire Cable**
1	Orange	N/A	N/A
2	Grey	N/A	N/A
3	Black	Ground	Ground
4	Green	SCL (Serial Clock)	SCK (Serial Clock)
5	Red	N/A	N/A
6	Yellow	SDA (Data In / Out)	SDI / SI (Serial Data In)
7	Brown	N/A	CS (Chip Select)
8	Blue	N/A	SDO / SO (Serial Data Out)

Table 3.3 - Interface Cable Data

PRELIMINARY

3.2.7 Custom Interface Solution

If you would like to create your own custom TMS-120 interface solution, you may purchase a cable without terminations on the target EEPROM side and connect it using any special connector, etc. that you choose. Spare (non-terminated) cables may be obtained from Triangle Micro Solutions (option P/N TMS-CBL). The length of a spare cable is seven feet (5'). Spare cables are terminated only at one end with the RJ-45 connector. The other end may be connected to your equipment however you choose. Use the information listed in table 3.3 above to determine how to connect the cable to your hardware.

Copyright © 2007 - Triangle Micro Solutions, Inc.

Figure 3.7 - Cable for Custom Interface Applications (option P/N TMS-CBL)

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.3 Software Configuration

3.3.1 EEPROM Tool Software Configuration

The EEPROM Tool software stores configuration parameters in the *Windows™ System Registry*. Each time the program is run it will reconfigure itself based on previous settings stored in the system registry. For first time users, there are built in defaults for all parameters except for the COM port. When you run the program for the first time, the configuration dialog box will be displayed. Be sure to select the COM port that the TMS-120 module will be connected to. Figure 4.1 shows the Configuration dialog box.

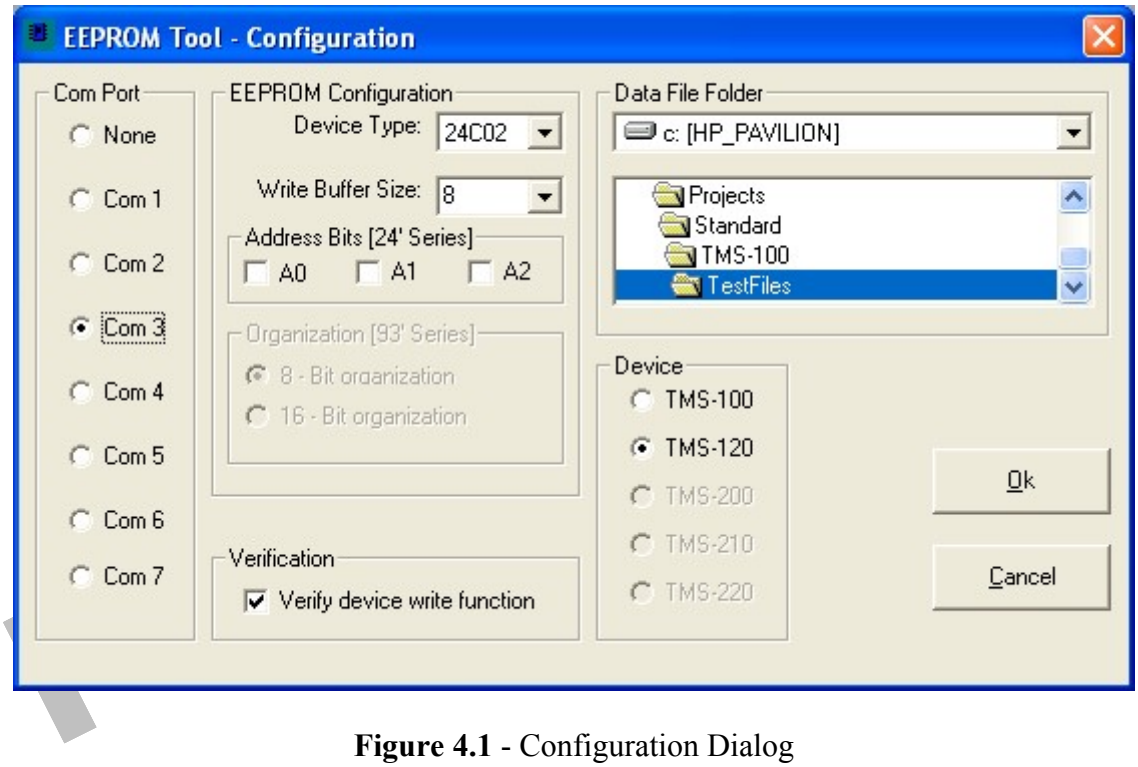


Figure 4.1 - Configuration Dialog

3.3.2 Communications Port

To communicate with the TMS-120, the proper communications port must be selected. This software supports COM1 through COM6. Although "None" is an available option, checking it will render the software unable to communicate with the TMS-120 Interface Module.

3.3.3 EEPROM Configuration

Use this control group to select the EEPROM device type, write buffer size, address bit settings, and organization.

3.3.4 Device type

From the list box, select TMS-120 from the list of available devices.

Copyright © 2007 - Triangle Micro Solutions, Inc.

3.3.5 Write Buffer Size

All serial EEPROM devices have a limit for the number of bytes that may be written at one time. This is usually called a "page write". The EEPROM Tool will automatically select a value for this parameter based on the device that you selected. The value selected automatically may not always be correct for the device you are trying to write to. This value may vary among different manufacturers. Check the data sheet for your specific EEPROM if you think there may be a problem with performing page writes. If the default parameter needs to be changed, select the correct value from the pull-down combo menu and select "ok".

NOTE

The maximum size allowed currently is 32 bytes. Some devices are designed to use 64 byte write buffers, but the TMS-120 is currently limited to 32 bytes at this time. Choosing a buffer size smaller than the maximum buffer allowed for the device selected is ok, but it will take more time to write data to the entire device.

3.3.6 Address Pins A0, A1, and A2 (24C' Series Only)

Some I^2C EEPROM devices have three pins labeled A0, A1 and A2. These pins are used to modify the address that the chip will respond to, thus allowing up to eight EEPROM devices to exist on the same I^2C bus. The EEPROM Tool software can be configured to work with any address configuration by clicking on the appropriate check boxes on the configuration dialog box. This control is enabled only when a 24C' series device has been selected.

NOTE

Do not check the A0, A1 or A2 box unless you are sure that the device that you want to read or write has that pin tied to Vcc. Check the schematic diagram of the circuit to see if the address pins are used.

3.3.7 Organization (93C' Series Only)

Most Microwire serial EEPROMs have an "organization" pin. When this pin is tied low (Vss), data is clocked out in 8-bit streams. When this pin is tied high (Vcc) or floating, data is clocked out in 16-bit streams. Check the schematic or use a voltmeter to determine the state of this pin. Select the proper organization for the device that you want to read or write. This control is enabled only when a 93C' series device has been selected. The 16-bit organization must also be made if the EEPROM does not have an organization pin, but is organized as a 16-bit part.

Verification

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

Check the "Verify device write function" box if you would like to perform verification when writing the contents of a file to an EEPROM. This defaults to the unchecked state, meaning that the action is not performed. When verification is enabled, writing an entire EEPROM takes longer to complete.

3.3.8 Data File Folder Selection

Data read from EEPROMS may be stored in files. To select the folder where these files will be stored, use the controls in the "Data File Folder" section of the configuration dialog box. You may select a drive and folder that you wish to store your data files into. The default directory is "C:\My Documents".

3.3.9 Exiting the Configuration Dialog

Press the ***Ok*** button to accept the changes that have been made to the program configuration. Press the ***Cancel*** button to exit the configuration menu without saving the changes.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

4 Using the TMS-120

4.1 Attaching the TMS-120 to a Target EEPROM

This section describes in detail how to connect the TMS-120 to your target EEPROM. The TMS-120 was designed to program EEPROM devices in circuit. Section 5.1.1 shows a typical connection to an I^2C (24C' series) EEPROM. Section 5.1.2 shows a typical connection to a SPI (25C' series) EEPROM and section 5.1.3 shows a typical connection to a Microwire (93C' series) EEPROM. These examples are typical, however pin functionality and package styles may vary. Please consult the data sheet for the specific device that you intend to use the TMS-120 with for actual pin functions. In the following examples we'll start with the assumption that the target device powered is off and the interface cable disconnected from the TMS-120 module (see warning).

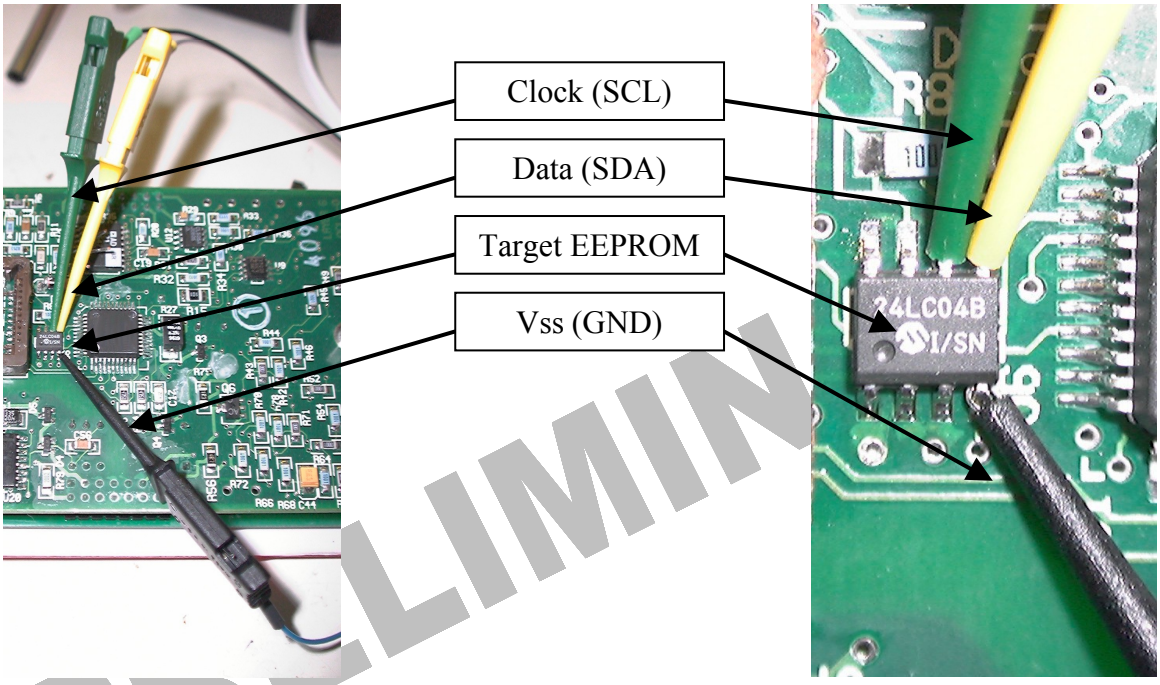
WARNING: Use caution when making electrical connections. Failure to connect the interface cable to the proper bus pins may result in damage to the TMS-120, the target EEPROM or your computer. Should you decide to connect the interface cable to the target device while the interface cable is connected to the TMS-120, or with power applied, you must assume all responsibility of damage.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

4.1.1 Typical I2C (24C' Series) Connection

Here is an example here of a typical connection to an I^2C (24C' series) EEPROM in a SOIC package. After attaching the test hooks to the EEPROM, check your connections thoroughly. Next plug the interface cable into the TMS-120 interface module using the RJ-45 plug. Finally, power up the target device containing the EEPROM. Start the EEPROM Tool software and you are ready to begin.



PRELIMIN

Figure 5.1 - Typical Connection an I^2C EEPROM (24C' Series)

Test Hook Color	Pin Function	Typical Pin Number (see note 4)
Black	Ground	3
Yellow	SDA (Data In / Out)	6
Green	SCL (Serial Clock)	4

Table 5.1 - Typical I^2C Interface Cable Connections

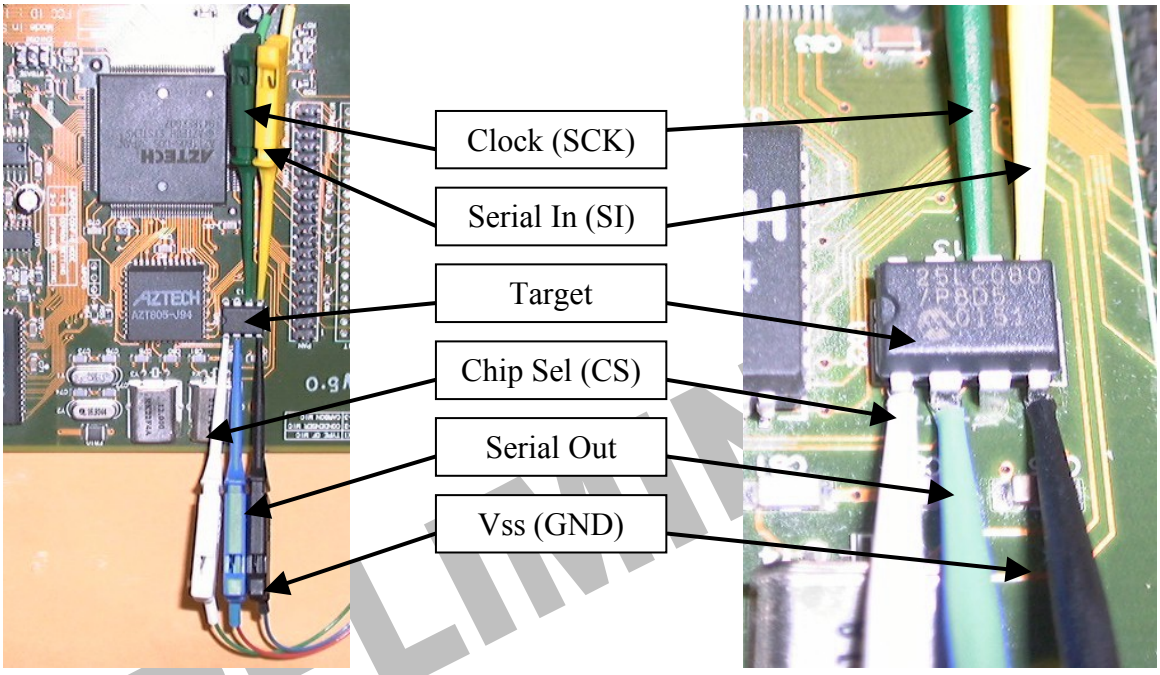
NOTES:

1. It is recommended that you always connect Vss (ground) first.
2. Make connections to target device with target power off
3. Visually inspect connections to verify that there are no adjacent pins shorted together.
4. Typical Pin Number for 8-pin DIP and SOIC packages. Verify pin numbers on the data sheet for your device before making connections.

Copyright © 2007 - Triangle Micro Solutions, Inc.

4.1.2 Typical SPI (25C' Series) Connection

Here is an example here of a typical connection to an SPI (25C' series) EEPROM in a DIP package. After attaching the test hooks to the EEPROM, check your connections thoroughly. Then plug the interface cable into the TMS-120 interface module using the RJ-45 plug. Finally, power up the target device containing the EEPROM. Start the EEPROM Tool software and you are ready to begin.



PRELIMINARY

Figure 5.2 - Typical Connection an SPI EEPROM (25C' Series)

Test Hook Color	Pin Function	Typical Pin Number (see note 4)
Black	Ground	3
Green	SCK (Serial Clock)	4
Yellow	SDI / SI (Serial Data In)	6
White (Brown Wire)	CS (Chip Select)	7
Blue	SDO / SO (Serial Data Out)	8

Table 5.2 - Typical SPI Interface Cable Connections

Copyright © 2007 - Triangle Micro Solutions, Inc.

NOTES:

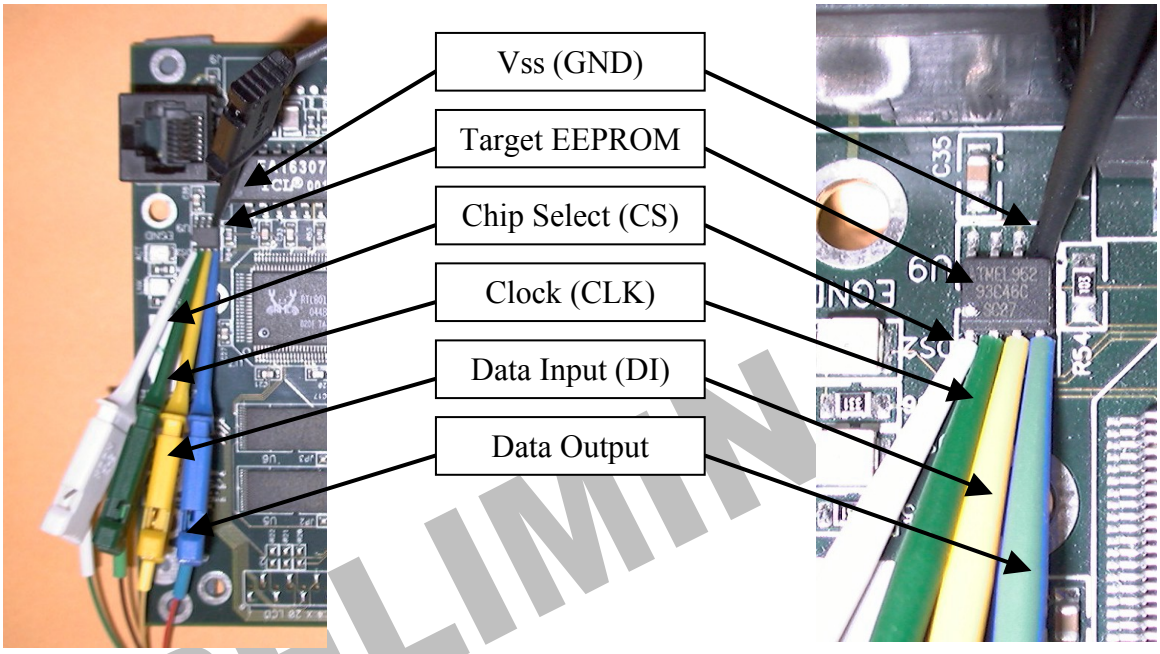
1. It is recommended that you always connect Vss (ground) first.
2. Make connections to target device with target power off.
3. Visually inspect connections to verify that there are no adjacent pins shorted together.
4. Typical Pin Number for 8-pin DIP and SOIC packages. Verify pin numbers on the data sheet for your device before making connections.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

4.1.3 Typical Microwire (93C' Series) Connection

Here is an example here of a typical connection to a Microwire (93C' series) EEPROM in a SOIC package. After attaching the test hooks to the EEPROM, check your connections thoroughly. Then plug the interface cable into the TMS-120 interface module using the RJ-45 plug. Finally, power up the target device containing the EEPROM. Start the EEPROM Tool software and you are ready to begin.



PRELIMIN

Figure 5.3 - Typical Connection a Microwire EEPROM (93C' Series)

Test Hook Color	Pin Function	Typical Pin Number (see note 4)
Black	Ground	3
Green	SCK (Serial Clock)	4
Yellow	SDI / SI (Serial Data In)	6
White (Brown Wire)	CS (Chip Select)	7
Blue	SDO / SO (Serial Data Out)	8

Table 5.3 - Typical Microwire Interface Cable Connections

Copyright © 2007 - Triangle Micro Solutions, Inc.

NOTES:

1. It is recommended that you always connect Vss (ground) first.
2. Make connections to target device with target power off
3. Visually inspect connections to verify that there are no adjacent pins shorted together.
4. Typical Pin Number for 8-pin DIP and SOIC packages. Verify pin numbers on the data sheet for your device before making connections.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5 Using the EEPROM Tool

From the ***START*** button, select ***Programs*** > ***EEPROM Tool***. Then select the ***EEPROM Tool*** icon. For convince, you may want to create a shortcut on your desktop. Figure 5.4 shows the main dialog window for the ***EEPROM Tool*** application. Each individual part of the main screen is labeled with a letter corresponding to a description witch follows in table 5.4. When the program is first started, it will attempt to communicate with the TMS-120 Interface Module. If it is successful, there will be a message on the status bar stating "Interface Module Connected". This indicates that the software has successfully communicated with the TMS-120 module and is ready for use. For ease of use, a tool-bar is located at the top of the main dialog window. Just click the appropriate icon to perform the related function. Each tool-bar icon is described in table 5.5.

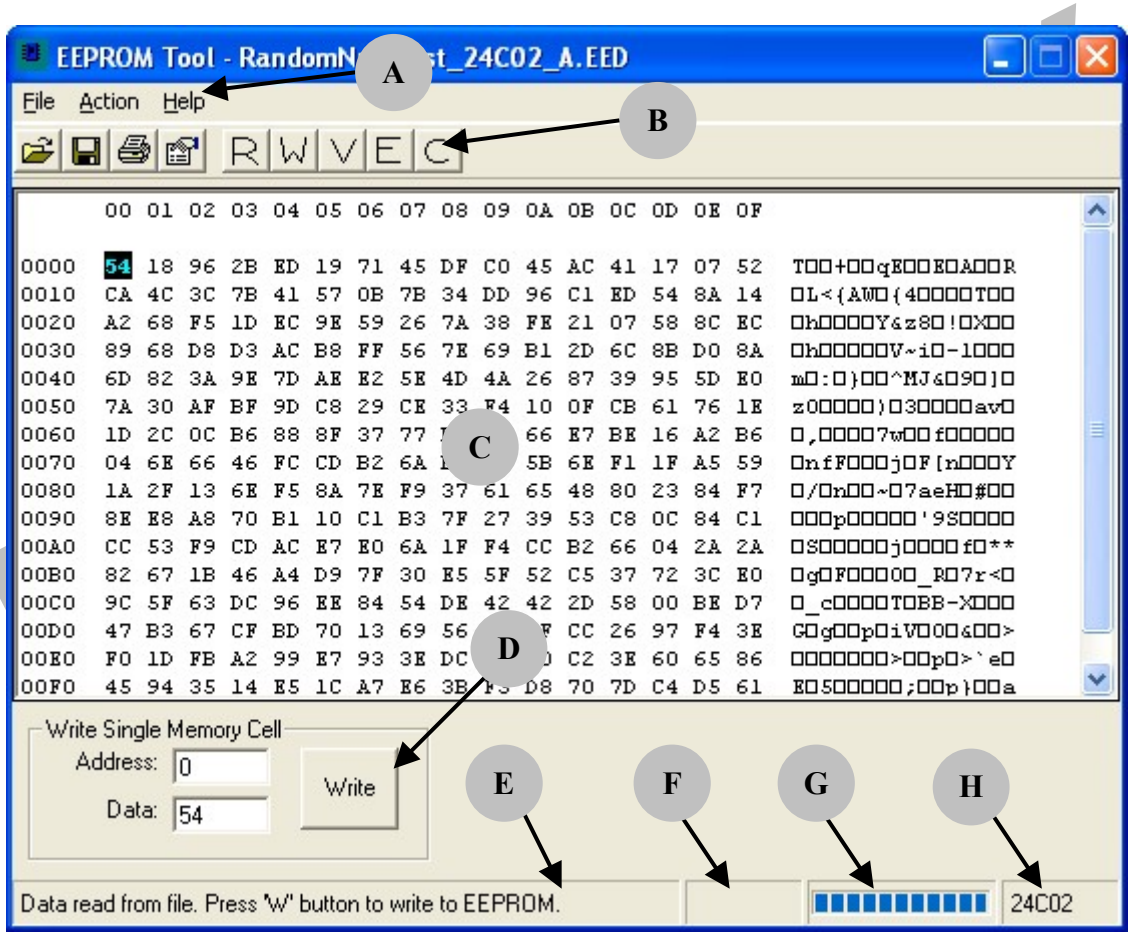


Figure 5.4 - The Main Dialog

Copyright © 2007 - Triangle Micro Solutions, Inc.

Item	Description
A	The drop-down menu bar.
B	The Tool Bar. See table 5.5 for a detailed description of each icon on the tool bar.
C	The Data Window. Currently selected data is shown in red.
D	Write Memory Cell. See section 5.5 for more information.
E	Program Status
F	Checksum.
G	Progress Bar. A visual indication of activity progress (0 - 100%).
H	Currently selected device I.D.

Table 5.4 - Main Dialog Features

Icon	Action	Description
	Open	Use the **Open** button to open EEPROM data files. Data from the file will be read into the program and displayed on the data gird in red.
	Save	Use the **Save** button to save the data currently in the data grid to an EEPROM data file (.EED extension). The type of EEPROM device that the program is configured for will be saved too.
	Print	Use the **Print** button to create a printed version of the data that is currently displayed on the data grid.
	Configure	Use the **Configure** button to bring up the program configuration dialog box. This is the same dialog box shown in figure 4.1.
	Read EEPROM	Use the **Read EEPROM** button to read a memory device's contents into the Data Window. The device must first be read before the data can be saved or before data can be changed.
	Write EEPROM	Use the **Write EEPROM** button to write the data from an EEPROM data file to an EEPROM device.
	Verify EEPROM	Use the **Verify EEPROM** button to compare the data stored in the data buffer with the actual EEPROM contents.
	Erase EEPROM	Use the **Erase EEPROM** button to erase the EEPROM. When the EEPROM is erased, all bits are set to "1".
	Clear	Use the **Clear** button to clear data from the grid and initialize the program for a new device.

Table 5.5 - Tool Bar Buttons and Descriptions

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.1 Reading Data from the EEPROM

To read data from a device, press the R icon on the toolbar or select ***Action*** | ***Read Device*** from the pull down menu bar. The status bar will show the progress being made. After all of the data has been completely read from the EEPROM, the data will be displayed in the data window. If you do not see data, check the status bar to see if there is an error condition indicated. After performing a read operation, the checksum field on the status bar will contain an updated checksum value. The checksum is a simple summation of all cell values read from the device.

5.2 Writing Data to a Single EEPROM Cell

At the bottom of the main dialog window there is a group of controls titled "Write Single Memory Cell". These controls are used to write data directly to a single EEPROM cell. The current address is shown in the *Address* text box. The data contained in the cell at the location is shown in the *Data* text box. You may select the address that you wish to write to in one of three ways. By navigating the data array using your computer keyboard (arrow keys, home key, end key), by clicking on an individual cell with the mouse cursor, or by directly entering the address into the Address text box (type in address and press the Enter key). The data currently stored at the selected address will be displayed in the Data text box. To change the data, enter the new value into the data text box and then click on the ***Write*** button. The new data will be written to the address shown. Values in the *Address* and *Data* text boxes are always shown in hexadecimal. Data should be entered in hexadecimal too.

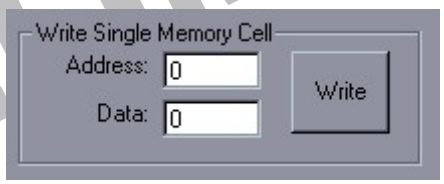


Figure 5.5 - Write Single Cell Controls (on Main Dialog)

PRELIMINARY

STOP **WARNING:** It is recommended that you save the contents of the EEPROM before making changes, especially if you are unsure of the consequences.

NOTE The EEPROM Tool software does not detect when a device is write protected. You must make sure that the write operation is enabled.

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.3 Verifying EEPROM Contents

Verification simply compares the contents of the EEPROM Tool Software's data buffer to the data in the EEPROM. To verify a device, press the V button on the toolbar or select ***Action*** | ***Verify Device*** from the pull down menu. If the verification process fails, the address is shown on the status bar along with the expected value and the value actually read. Verification may be automatically performed when writing an entire EEPROM from a file (see section 5.10) if the option "Verify device write function" is checked in the configuration dialog box. The checksum field on the status bar is updated upon successfully completing the verification.

5.4 Erasing an EEPROM

All supported serial EEPROM devices can be erased. To erase an EEPROM, press the E button on the toolbar or select ***Action*** | ***Erase*** from the pull down menu. Before proceeding, the dialog box shown in figure 5.6 will be displayed. This is to inform the user that the erase procedure is about to begin and that all connections to the target device should be made before proceeding. Click ***Ok*** to continue. Click ***Cancel*** to abort the erase function.

Figure 5.6 - Erase Acknowledgement Dialog

When the device is erased, all bits will be written to "1". After the erase procedure is completed, the data window in the main dialog box will be clear. The dialog box shown in figure 5.7 will be displayed showing how many EEPROMs you have erased thus far and asking you would like to erase another EEPROM. Press ***Yes*** to erase another and ***No*** to exit the erase function all together. If you select ***Yes***, you will be presented with the dialog box shown in figure 5.6 again and the cycle will continue until you press ***No*** when the dialog box shown in figure 5.7 is presented.

Figure 5.7 - Erase Continuation Dialog

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.5 Exporting Data to Disk File

You may want to store the data in your EEPROM to a disk file. To do so, first read the data from your EEPROM into the program. Then select ***File | Save*** or ***File | Save As*** from the drop-down menu bar. Then enter (or select) a file name when the file dialog box appears. The EEPROM data files created with this program should always carry the .EED extension. The specific device, which this data was read from, is included in the file so that it may be compared with the device that the program was configured for when loaded back into the program.

5.6 Coping EEPROMs

You may copy the contents of one EEPROM to another by first reading the EEPROM (see section 5.3 above) then disconnect the TMS-120 from the source EEPROM and connect it to the target EEPROM. Then write the contents to the target device (see section 5.11 below). Multiple copies may be made simply by removing the old target device and replacing it with a new target device and writing the data to the new target.

5.7 Clearing the Data Window and Initializing the Program

When you want to read a new device, you may want to first clear the old EEPROM data from the data window. Press the icon on the toolbar or select ***Action | Clear Data Window*** from the drop-down menu bar.

5.8 Importing Data from Disk File

Once data has been stored in a disk file, it may be retrieved later and programmed into other devices. There are two ways that a data file may be opened. To open a file from the tool bar, press the **Open** icon on the toolbar, or from the drop-down menu bar, select ***File | Open*** and then select the desired file. Data from the file will be displayed in the Data Window.

The default file extension is .EED (EEPROM Tool Data Format). However, you may import data from any file, so long as there is sufficient room in the EEPROM to store the data. To open another file type, select **All Files (*.*)** from the **Files of type:** selection area of the **Open** dialog box. When opening non-EED file types, you will be prompted for an address to begin loading the data at. This defaults to 0. You may enter any valid address for the EEPROM being used, but if there is insufficient space from the address that you enter to the last address in the EEPROM to hold the contents of the file, you will get an error message.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.9 Writing an Entire EEPROM at Once

This feature writes an entire EEPROM from data that has been imported from a file or data that has been read from another EEPROM. Data may be written to one or more EEPROM devices. Once data has been loaded, you may write it to an EEPROM by selecting ***Action*** | ***Write*** from the drop-down menu or by pressing the icon on the toolbar. Upon initiating the write action, the dialog box shown in figure 5.8 will be displayed. This is to inform the user that the erase procedure is about to begin and that all connections to the target device should be made before proceeding. Click ***Ok*** to continue. Click ***Cancel*** to abort the erase function.

WARNING: It is recommended that you save the contents of the EEPROM before making changes, especially if you are unsure of the consequences.

Figure 5.8 - Write Acknowledgement Dialog

After the write procedure is completed, the dialog box shown in figure 5.9 will be displayed showing the number of EEPROMs you have written thus far and asking you would like to write another EEPROM. Press ***Yes*** to program another and ***No*** to exit the Write function all together. If you select ***Yes***, you will be presented with the dialog box shown in figure 5.8 again and the cycle will continue until you press ***No*** when the dialog box shown in figure 5.9 is presented.

Figure 5.9 - Write Continuation Dialog

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.10 Printing Data

Sometimes you may need a hardcopy of data stored in an EEPROM. Once data has been read from the EEPROM and displayed in the data window, you can print it. There are two ways to print the data. From the drop-down menu, select ***File*** | ***Print…*** or with the mouse, click the printer icon located on the toolbar. After making the print selection, you will be presented with the "Print Notes" dialog box shown in figure 5.10 below. This feature allows you to add notes to the printed data as well as the device type (EEPROM part number) and the current date.

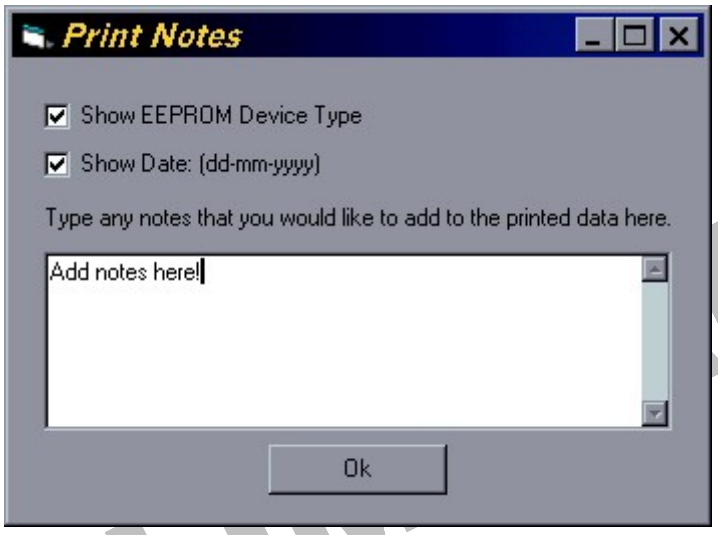

Print Notes
Show EEPROM Device Type
Show Date: (dd-mm-yyyy)
Type any notes that you would like to add to the printed data here.
Add notes here!
Ok

Figure 5.10 - Print Notes Dialog

After adding notes, press the ***Ok*** button to proceed. You will then be presented with the familiar windows print dialog box. From there, press the ***Ok*** button to print the data. Press the ***Cancel*** button if you decide that you do not want to print the data.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

5.11 The SPI EEPROM Status Register

The SPI EEPROM devices (25C' Series) contain a status register. The status register controls (in conjunction with the write protect pin) write protection for the device. The read or write bits in the status register, select ***Action | SPI Status Register*** from the drop-down menu bar. This selection is only valid when an SPI device is selected (otherwise it will be grayed out). When this function is selected, the dialog box shown in figure 5.11 will be displayed. All eight bits of the status register are shown, but some are read only or not used (they are shown grayed out). The main bits of interest are BP0 (bit 2) and BP1 (bit 3) which are the block protect bits. When set, certain portions of the EEPROM are write protected. Bit 7 may or may not be supported by the specific device being used. See the data sheet for your specific device for more information.

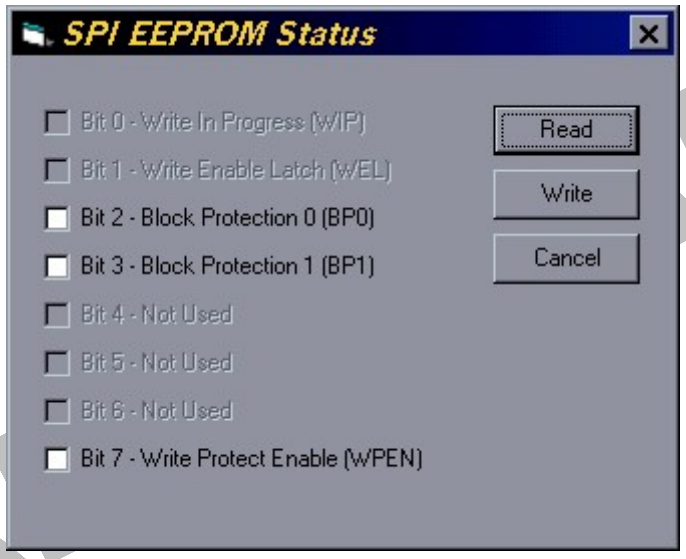

SPI EEPROM Status
Bit 0 - Write In Progress (WIP)
Bit 1 - Write Enable Latch (WEL)
Bit 2 - Block Protection 0 (BP0)
Bit 3 - Block Protection 1 (BP1)
Bit 4 - Not Used
Bit 5 - Not Used
Bit 6 - Not Used
Bit 7 - Write Protect Enable (WPEN)
Read
Write
Cancel

Figure 5.11 - SPI EEPROM Status Register

Press the "Read" button to read the current state of the Status Register bits. Make changes as needed by checking (or un-checking) the appropriate boxes and press the "Write" button. To return to the main dialog, press the "Cancel" button. Note that when reading or writing the status register, operational status will be displayed on the status bar of the main dialog.

Copyright © 2007 - Triangle Micro Solutions, Inc.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

6 Troubleshooting

6.1 Verify Interface (Data Communications Test)

To verify that the TMS-120 interface module is working, select ***Action | Verify Interface*** from the drop-down menu. You should see the dialog box shown in figure 6.1 below. If so, you have verified that the TMS-120 Interface Module is working and that the PC can communicate with it over the serial port (RS-232 com port).

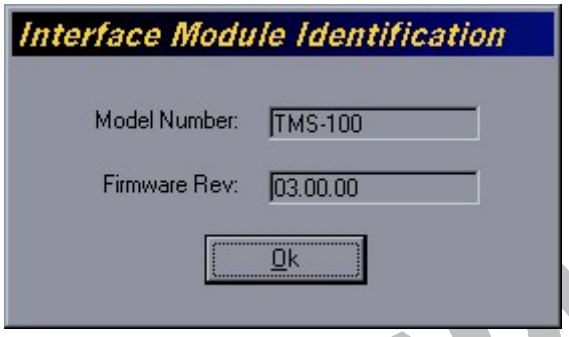


Figure 6.1 - Interface Module Identification Dialog

If you see the message "ERROR: Communications timed out" on the status bar of the main dialog, communications between the TMS-120 Interface Module and the PC are not working. See section 6.2.2 for trouble shooting tips.

6.2 The Status Bar

The status bar at the bottom of the main dialog window will show the current status or progress for a specific task. It may contain error messages or other helpful information that can be used to troubleshoot problems.

6.2.1 General I/O Errors

General I/O errors may occur when performing file, disk or printer I/O functions. There numerous errors that could result from various systems failures. Here are some possible error messages that may be seen. Below each error are some trouble shooting tips.

- **Error: File not found**
  The supplied file name does not exist. Check file name and try again.

- **Error: File already exists**
  Choose another file name. The supplied name is in use already.

- **Error: Device I/O error**
  Device failure.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

- **Error: Disk full**
  Create additional space or save file to another disk.

- **Error: Printer error**
  Check cables, power connections. Try again.

6.2.2 Communications Errors

Occasionally the serial communications between the PC and the TMS-120 may be corrupted. This might happen for various reasons. Since the EEPROM Tool software will attempt to send the message several times before giving up, you should never encounter the majority of these error messages.

- **Error: Communications timed out.**
  The TMS-120 failed to respond to the command sent from the PC within the allotted time frame. Here are some things to check:

  - Program configuration (see section 4.0). Make sure that the serial port selected is the same serial port that the TMS-120 is connected to.

  - Check your physical connection of the TMS-120 module to your serial port. Make sure that it is completely plugged into the DB25 connector.

  - If you are using a 9-pin to 25-pin converter or an extension cable, be sure that it is plugged in completely.

  - Make sure you are not using a Null Modem cable or adapter in your serial port connection.

- **Error: No ACK seen.**
  The serial EEPORM (24C' series) failed to ACK when expected. Check the following:

  - Verify SCL and SDA connections are not reversed.

  - Verify device type matches EEPROM Tool device selection.

  - Verify A0, A1, and A2 address bits are set correctly in the configuration dialog.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

You will not likely see the errors in described in the remainder of this section because the TMS-120 will automatically retransmit a frame that failed due to one of these errors.

- **Error: Frame length too short.**
  The received frame was shorter than expected.

- **Error: Invalid start character.**
  The first character in the frame was not SOH as required.

- **Error: Invalid command.**
  The command received is not a recognized command

- **Error: Invalid number of data bytes.**
  The number of data bytes encountered was less than expected.

- **Error: Invalid checksum.**
  The serial data frame checksum check failed. This is likely due to a serial data corruption.

- **Error: Invalid frame end character.**
  The frame was not terminated with the EOT character as required.

6.2.3 Operational Errors

Several operational errors might occur while using the EEPROM Tool software. Operational error messages are displayed as a pop-up message box to alert the user that a requested action can not be performed. These require that the user acknowledge them by pressing the ***Ok*** button in the message box to clear them..

- **Error: File contents can not fit at selected location.**
  An attempt was made to load data from a file into buffer space that is too small to contain all of the data in the file. This may occur when importing data from a non-EED file type. Check the location address, the EEPROM size and the size of the file being loaded.

- **Error: Device type does not agree with configured device type. Please check device configuration and try again.**
  An attempt was made to load data from a .EED file that was saved from a different device type from the currently selected device type. For example, if the file contained data saved from a 24C08 and the currently selected device is 24C16, this error would be generated.

- **Error: Data must be read first.**
  An attempt was made to print data from the program before data has been read. You must read the contents of the EEPROM first before attempting to print data.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

- **Error: Can only save data to .EED file format. Please use Save As and select an EED extension.**

  An attempt was made to save data to a file when the file extension was not .EED. All saved files must be saved with the EED extension. In the

- **Error: No data has been read from device.**

  An attempt was made to save data to a file when the data buffer was empty (i.e.: no data has been read from the EEPROM). Read the EEPROM first before attempting to save data to a file.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

7 Communication Protocol

7.1 Protocol Basics

This section describes the protocol that host PC must follow in order to communicate with the TMS-120 interface module. This information is included for those who wish to integrate communication with a TMS-120 into their own application. Communications between a PC and the TMS-120 are half-duplex with the PC being the host (i.e. the PC always initiates communications). The PC serial port should be configured as follows: 9600 Baud, No Parity, 8 Data bits, and 1 Stop Bit. Commands frames are transmitted from the host via the RS-232 port to the TMS-120. Command frames consist of byte oriented fields as described below. The TMS-120 will respond after completing the requested task, usually in less than 500mS. This general format is illustrated in section 7.2 below. In this section, numbers prefixed with '0x' indicate hexadecimal.

7.2 General Command Protocol

The general command protocol allows for up to 256 commands with a variable length data field for both the host to slave and slave to host command formats. The frame formats show the number of bytes in a particular field in brackets [ ].

Host to device frame format:

SOH	C	N	D [0 - 64]	CS	EOT

SOH = Start of frame delimiter (always 0x01)
C = Command number. Supports up to 256 commands in the range of 00 - FFH (256).
N = Number of bytes to follow (2 - 66) - CS and EOT are included in this count.
D = Data field, may contain zero to sixty-four bytes of data.
CS = Longitudinal parity check of bytes from command to last data byte (x'oring).
EOT = End of frame delimiter (always 0x04)

Device to host frame format:

SOH	C	S	N	D [0 - 64]	CS	EOT

SOH = Frame start character (always 0x01)
C = Command number. Supports up to 256 commands in the range of 00 - FFH (256).
S = Frame status. See section 4.2.1 below for list of general status bytes

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

N = Number of bytes to follow (2 - 66) - CS and EOT are included in this count.

D = Data field, may contain zero to sixty-four bytes of data.

CS = Longitudinal parity check of bytes from command to last data byte (x'oring).

EOT = Frame end character (always 0x04)

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

7.3 Status Codes

Status codes are in the range of 00 to 07. Status codes are contained in the Device to Host frame (see section 7.6 below). A status code of zero indicates that the operation was successful. If the status code is greater than zero, the *number of data bytes* shall be 0 (no data should be included) followed by the [EOT].

00 - No error - operation successful
01 - Invalid frame length
02 - Invalid start character (first character not SOH - 0x01)
03 - Invalid command (or unsupported command)
04 - Invalid number of data bytes received
05 - Reserved
06 - Invalid checksum
07 - Invalid frame end character (last character not EOT - 0x04)

Checksum Calculation
The checksum field contains the longitudinal parity check of all the bytes between the SOH and the last byte in the data field. This type of checksum is calculated by exclusive-or'ing each byte in the sequence.

Fixed vs. Variable Length Fields
This protocol allows variable length fields. Unless noted in brackets [ ], a field will always be one byte in length. For example, if a field has a fixed size of three bytes, a [3] will be shown in the frame description. An example of a variable length data field would look like this [2 - 20].
Command 0x00 - Device Identification
This command provides a means of verifying that the device is powered, communicating, identifies the firmware version and is ready to use. The response data provides a specific identification for the device. Any application should send this command first to verify that the device is active.

PRELIMINARY

Host to device frame format:

SOH	C	N	CS	EOT

SOH Start of frame delimiter
C Command 00 Hex
N Number of bytes that follow must be 02H
CS Checksum is 02 Hex
EOT End of frame delimiter

Device to host frame format:

Copyright © 2007 - Triangle Micro Solutions, Inc.

SOH	C	S	N	HWC [3]	FWC [3]	CS	EOT
Example: TMS-120 = 0x78 Firmware Revision 1.0.0				00 00 78	01 00 00		

SOH Start of frame delimiter
C Command is 00H
S Status Code (See section 7.3)
N Number of bytes that follow must be 08H
HWC Hardware identification code
FWC Firmware identification code
CS Checksum
EOT End of frame delimiter

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

7.4 Command 0x10 - Read Serial EEPROM Data

Host to device frame format:

SOH	C	N	BLK	ADDMSB	ADDLSB	NDB	CS	EOT

SOH Start of frame delimiter – ASCII 0x01
C Command 10 hex
N Number of bytes to follow including CS and EOT
BLK Block control byte. Contains hardware address bits and page bits for 24C00-24C16 devices. Also contains a bit, which indicates if the ADDMSB is used, or not. See table 5.1

Bit 0-2:
The meaning of these bits is dependant on the device type being used. For I^2C Bus EEPROMs, these bits are used to indicate the state of the extra address bits (A0, A1, and A2). These are also used as page bits on some devices.

I^2C Bus EEPROMS (24C series)	**Bit 2**	**Bit 1**	**Bit 0**
	A2 / P2	A1 / P1	A0 / P0

For Microwire devices, these bits are used to communicate the specific device being used to the interface module (TMS-120). This is necessary so that the interface module knows how many clocks are requires to be sent, since this varies from device to device.

Microwire EEPROM P/N	**Bit 2**	**Bit 1**	**Bit 0**
93C06	0	0	0
93C46	0	0	1
93C56	0	1	0
93C66	0	1	1
93C76	1	0	0
93C86	1	0	1
N/U	1	1	0
N/U	1	1	1

Bit 3:
0 – ADDMSB is not used
1 – ADDMSB is used

Bit 4-5:

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

Device bus style

Bus Style	Bit 5	Bit 4
I2C Bus (24XX Series)	0	0
SPI (25XX Series)	0	1
Microwire (8-bit) (93XX Series)	1	0
Microwire (16-bit) (93XX Series)	1	1

Bit 6-7:
Not used

ADDMSB	The most significant address byte. If CB (Bit 3) is set, this field contains the Address MSB.
ADDLSB	Least significant byte of address
NDB	Number of data bytes requested. This can be any value from 1 to n; where n is the number **Page Write Buffer Size.**
CS	Check sum
EOT	End of frame delimiter – ASCII 0x04

Device to host frame:

SOH	C	S	N	D [0-64]	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 10 Hex
S	Status byte [see sec. 7.3 for possible errors]
N	Number of bytes to follow including CS and EOT
D	The data read from the device, this is a variable length field [0 to 64 bytes]. This field will not normally contain "0" unless there was an error encountered.
CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

NOTE: If SS is greater than 0 then NN will be 2 and no data bytes would follow.

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

7.5 Command 0x11 - Write Serial EEPROM Data

Host to device frame format:

SOH	C	N	BLK	ADDMSB	ADDLSB	DF [1-64]	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 11 Hex
N	Number of bytes that follow
BLK	Control Block - bit mapped option list. See command 0x10.
ADDMSB	Most significant byte of address
ADDLSB	Least significant byte of address
DF	The data written to the device, this is a variable length field [1 to 64 bytes].
CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

Device to host frame format:

SOH	C	S	N	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 11 Hex
S	Status [see sec. 7.3 for possible errors]
N	Number of bytes that follow including CS and EOT
CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

PRELIMINARY

7.6 Command 0x12 - Read Status Register of SPI EEPROM

Use this command to retrieve the contents of the SPI EEPROM Status Register.

Host to device frame format:

SOH	C	N	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 12 Hex
N	Number of bytes that follow (2)

Copyright © 2007 - Triangle Micro Solutions, Inc.

CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

PRELIMINARY

Copyright © 2007 - Triangle Micro Solutions, Inc.

Device to host frame format:

SOH	C	S	N	D	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 12 Hex
S	Status [see sec. 7.3 for possible errors]
N	Number of bytes that follow including CS and EOT (3 or 2)
D	Data (SPI EEPROM Status)
CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

7.7 Command 0x13 - Write Status Register of SPI EEPROM

Host to device frame format:

SOH	C	N	D	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 13 Hex
N	Number of bytes that follow (3)
D	Data, the new value for status register.
CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

PRELIMINARY

Device to host frame format:

EOT	C	S	N	CS	EOT

SOH	Start of frame delimiter - ASCII 0x01
C	Command 11 Hex
S	Status [see sec. 7.3 for possible errors]
N	Number of bytes that follow including CS and EOT
CS	Checksum
EOT	End of frame delimiter - ASCII 0x04

End of Document

Copyright © 2007 - Triangle Micro Solutions, Inc.